

[文章编号] 1003- 4684(2008) 02- 0044- 03

基于 VC 的 STL 文件读取

张贞贞¹, 陈定方²

(1 武汉理工大学计算机科学学院, 湖北 武汉 430063; 2 武汉理工大学物流工程学院, 湖北 武汉 430063)

[摘 要] 就 CAD 系统较常用文件格式 STL 的快速读取问题展开讨论, 分析了 STL 两种文件格式的特点, 并讨论了冗余数据的消除.

[关键词] STL 文件; ASCII; 二进制

[中图分类号] TP391

[文献标识码]: A

STL (Stereo Lithography interface specification) 文件格式^[1]最初出现于 1989 年美国 3D SYSTEM 公司生产的 SLA 快速成型系统, 是一种应用于 CAD 模型与成型系统之间数据转换的文件格式, 现在已为大多数 CAD 系统和快速成型系统制造商所接受和采用, 并且已经在快速成型技术领域中被誉为工业标准. 本文旨在研究如何高效率读取 STL 文件.

1 STL 文件格式的结构

STL 类似于实体数据模型的表面有限元网格划分. STL 模型的数据通过给出组成三角形法向量的 3 个分量及三角形的 3 个顶点坐标来实现, STL 文件记载了组成实体模型的所有三角形面片的法向量数据和顶点坐标数据, 有文本文件 (ASCII) 和二进制文件 (BINARY) 两种格式.

1.1 STL 的 ASCII 文件格式

ASCII 码格式 STL 文件逐行给出三角面片的几何信息, 每一行以 1 个或 2 个关键字开头. 在 STL 文件中的三角面片的信息单元 facet 是一个带矢量方向的三角面片, STL 模型就是由一组这样的三角面片构成. 在一个 STL 文件中, 每一个 facet 由 7 行数据组成, 第 1 行是三角面片指向实体外部的法向矢量数据, 第 2 行说明随后的第 3, 4, 5 行数据分别是三角面片的 3 个顶点信息, 沿指向实体外部的法向矢量 (第 1 行数据) 方向逆时针排列.

ASCII^[2] 格式的 STL 文件结构如下:

```
solid filename stl                                // 文件路径及文件名
facet normal x y z                                // 三角面片法向量的 3 个分量值
outer loop
vertex x y z                                       // 三角面片第一个顶点的坐标
vertex x y z                                       // 三角面片第二个顶点的坐标
vertex x y z                                       // 三角面片第三个顶点的坐标
endloop
endfacet                                           // 第一个三角面片定义完毕
.....
.....
.....
endsolid filename stl                             // 整个文件结束
```

1.2 STL 的二进制文件格式

二进制 STL 文件用固定的字节数来给出三角面片的几何信息. 文件的起始 80 字节是文件头存储零件名, 可以放入任何文字信息; 紧随着用 4 个字节的整数来描述实体的三角面片个数, 后面的内容就是逐个给出每个三角面片的几何信息. 每个三角面片占用固定的 50 字节, 它们依次是 3 个 4 字节浮点数, 用来描述三角面片的法矢量; 3 个 4 字节浮点数, 用来描述第 1 个顶点的坐标; 3 个 4 字节浮点数, 用来描述第 2 个顶点的坐标; 3 个 4 字节浮点数, 用来描述第 3 个顶点的坐标, 每个三角面片的最后 2 个字节用来描述三角面片的属性信息 (包括颜色属性等) 暂时没有用. 一个二进制 STL 文件的大小为三角形面片数乘以 50 再加上 84 个字节.

比较这 2 种方式可知, 二进制格式文件较小 (通常是 ASCII 码格式的 1/5)^[3], 节省存储空间, 但 ASCII 码格式的文件可读性更强, 能被人工识别并被修改.

[收稿日期] 2007- 12- 01

[作者简介] 张贞贞 (1982-), 女, 湖北枣阳人, 武汉理工大学硕士研究生, 研究方向: 计算机辅助设计与仿真.


```
else{
if(q == NULL){head = v;}
else{
if(q->next == p){q->next = v;}
if(q->x_same == p){q->x_same = v;}
if(q->y_same == p){q->y_same = v;}
v->y_same = p;v->x_same = p->x_same;
p->x_same = NULL;v->next = p->next;
p->next = NULL;return v;}}}
if(p == NULL){q->y_same = v; return v;}
else{// 在 X 坐标值相同的链中查找并插入点 v
if(v->y > p->y){q = p; p = p->x_same;}
else{
if(q == NULL){head = v;}
else{
if(q->next == p){q->next = v;}
if(q->x_same == p){q->x_same = v;}}
v->x_same = p;v->next = p->next;
p->next = NULL;return v;}}}
if(p == NULL){q->x_same = v; return v;}
else{// 在 X, Y, Z 坐标值均不同的链中查找并插入点 v
if(v->x > p->x){q = p; p = p->next;}
else{
if(q == NULL){head = v;}
else{q->next = v;}
v->next = p;return v;}}}
if(p == NULL){q->next = v; return v;}}}
```

然后读取 ASCII 形式的 STL 文件, 存储文件信息, 具体算法如下:

```
BOOL LoadSTLFile(LPCTSTR stlfile){
FILE* file;
if((file = fopen(stlfile, "r")) == NULL) return false;
char str[80];
vertex* v1, v2, v3;
facet* f;
while(fscanf(file, "%s", str) == 1){
if(strncmp(str, "normal", 6) == 0){
v1= (vertex*) malloc(sizeof(vertex));
v2= (vertex*) malloc(sizeof(vertex));
v3= (vertex*) malloc(sizeof(vertex));
f= (facet*) malloc(sizeof(facet));
// 读取面片的法向量
fscanf(file, "%lf %lf %lf", &(f->x), &(f->y), &(f->z));
fscanf(file, "%*s %*s");
// 读取面片的三个顶点信息到 v1, v2, v3
fscanf(file, "%*s %lf %lf %lf", &(v1->x), &(v1->y), &(v1->z));
fscanf(file, "%*s %lf %lf %lf", &(v2->x), &(v2->y), &(v2->z));
fscanf(file, "%*s %lf %lf %lf", &(v3->x), &(v3->y), &(v3->z));
// 分别判断 v1, v2, v3 是否在点表中出现过并完成面表到点表的映射
f->vetex1= Sort(v1, head);
f->vetex1= Sort(v2, head);
f->vetex1= Sort(v3, head);
m_facet. Add(f);}}}
```

最后, 通过顺序读取面表 m_facet 可以获取所

有三角面片信息, 在 VC 中利用 OpenGL^[5] 来实现三维图形的显示. 具体算法如下:

```
for (int i = 0; i < number; i+ ){
glBegin( GL_TRIANGLES);
glNormal3f( m_triangle[i].x, m_triangle[i].y, m_triangle[i].z);
glVertex3f( m_triangle[i].vertex1->x, m_triangle[i].vertex1->y, m_triangle[i].vertex1->z);
glVertex3f( m_triangle[i].vertex2->x, m_triangle[i].vertex2->y, m_triangle[i].vertex2->z);
glVertex3f( m_triangle[i].vertex3->x, m_triangle[i].vertex3->y, m_triangle[i].vertex3->z);
glEnd();
}
```

函数 glBegin^[5] 实用参数 GL_TRIANGLES 将产生实体造型效果, 即真实感图形显示.

3 结语

STL 文件中每个顶点通常被重用 6 次左右, 因而文件数据重复很多. 通过消除冗余顶点信息可以极大地减少存储空间, 提高对 STL 文件的读取和存储效率, 为三维物体的建模提供了方便. 此外, 采用多重链表形式存储定点信息在消除冗余顶点的同时, 使坐标值相同的点连在一起, 并且按照坐标值的升序排列, 减少了在查找新的顶点是否在点表中存在时的比较次数, 提高了查找效率.

[参 考 文 献]

[1] 张红玲. STL 格式实体真实感图形显示工具[J]. 电脑学习, 2000(4): 31- 32.
[2] 王清辉, 王 彪. Visual C+ + CAD 应用程序开发技术[M]. 北京: 机械工业出版社, 2003.
[3] 卫 炜, 周来水, 张丽艳. 海量 STL 文件的快速读取与显示[J]. 机械科学与技术, 2006, 25(6): 935- 938.
[4] 严蔚敏, 吴伟民. 数据结构[M]. 北京: 清华大学出版社, 2007.
[5] Shreiner D, Woo M, Neider J, et al. OpenGL 编程指南[M]. 徐 波译. 北京: 人民邮电出版社, 2005.

(下转第 73 页)

[参 考 文 献]

[1] 申忠如, 郭福田, 丁 晖. 现代测试技术与系统设计 [M]. 西安交通大学出版社, 2006: 102– 106.

[2] 赵浪涛. BP 神经网络 PID 控制器研究[J]. 兰州工业高等专科学校学报, 2007, 14(2): 34– 35.

[3] 凌振宝, 王 君, 朱凯光. 数字温度传感器在热电偶冷端温度补偿中的应用[J]. 传感器技术, 2003, 22(6): 45 – 46.

The Research on the Intelligent Instrument for Measuring Temperature Based on PID Neural Network

ZHU Ke¹, LIU Hong-li¹, ZHEN Yuyun^{1,2}

(1 The School of Automation, Wuhan Univ. of Technology, Wuhan 430063, China;

2 Institute of Intelligent Manufacturing, Wuhan Univ. of Technology, Wuhan 430063, China)

Abstract: Thermocouple is widely applied in industry temperature measurement, but it is not a linear relation between its temperature and potential, which limits the precision of measuring temperature. In order to solve the problem of the accuracy of thermocouple temperature calculation, the hardware circuit of Intelligent Instrument is designed with feed forward correction function of the PID Neural Network. The result shows that this computing method has good reliability and practicability.

Keywords: thermocouple; PID neural network; intelligent instrument for measuring temperature

[责任编辑: 张 众]

(上接第 46 页)

STL File Loading Based On VC

ZHANG Zhen-zhen¹, CHEN Ding-fang²

(1 School of Computer Science, Wuhan Univ. of Technology, Wuhan 430063, China;

2 School of Logistics Engine, Wuhan Univ. of Technology, Wuhan 430063, China)

Abstract: STL has been the industry standard in the field of rapid prototype technology. How to load and visualize the STL file rapidly with deleting redundancy vertex in STL files is discussed in this paper in the aspects of STL rapid processing of common file formats and the analysis of two STL file formats.

Keywords: STL; ASCII; binary

[责任编辑: 张 众]